

CLAIMS

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Previously Presented) A machine-implemented system that facilitates management of an application or service, comprising:
 - an application or service for installation on the system; and
 - an attribution component that facilitates attributing selected parts of code of the application or service with management information;where the system uses the management information to manage the installed application or service.
2. (Original) The system of claim 1, a software tool is applied to the attributed code of the application or service to expose the management information.
3. (Original) The system of claim 1, the management information is exposed and used to generate a manifest.
4. (Previously Presented) The system of claim 1, the attributed code of the application or service indicates at least one subset of types within one or more components of the application or service that should be exposed and how the subset of types should be identified.
5. (Original) The system of claim 1, the management information is exposed from at least one of an in-process provider and a decoupled provider.

6. (Original) The system of claim 1, the management information is exposed from a decoupled provider, which attributed code of the decoupled provider includes at least one of a register call at startup and an unregister call at shutdown.

7. (Original) The system of claim 1, a catalog is created of all available instrumentation data of the system, wherein the catalog is browsed and used to discover a particular instrumentation point.

8. (Original) The system of claim 1, at runtime, management information is retrieved by identifying the associated management information within a catalog of all management information of the system, and following the associated management information to the corresponding application or service.

9. (Original) The system of claim 8, for an in-process provider at runtime, the component associated with the management information, is loaded and invoked.

10. (Original) The system of claim 8, for a decoupled provider at runtime, the management information is used with information provided by a register call to locate a corresponding running process, to connect to the running process, and to locate a subcomponent within the running process that is associated with the management information.

11. (Original) The system of claim 1, the management information includes a probe attribute that is used to indicate that a member of a type is a probe.

12. (Original) The system of claim 11, the type is decorated with a folder attribute.

13. (Original) The system of claim 1, the management information includes health information that is exposed from an in-process provider.

14. (Original) The system of claim 1, the management information includes health information that indicates health of the application or service.

15. (Original) The system of claim 1, the management information is identified within the attributed application or service using a uniform resource identifier.

16. (Original) The system of claim 1, the management information includes execution information that indicates when the management information should be executed.

17. (Original) The system of claim 1, the management information is exposed from a data source that includes at least one of hardware, software application, and an operating system.

18. (Original) The system of claim 1, the management information includes class definitions that are exposed to a management component.

19. (Original) The system of claim 1, the class definitions are described in a managed object format.

20. (Original) A computer system according to claim 1.

21. (Original) A computer-readable medium having computer-executable instructions that embodies the system of claim 1.

22. (Previously Presented) A method of managing an application or service, comprising:

receiving the application or service for installation on a system;
attributing selected parts of code of the application or service with management information;
exposing the management information to a management system; and
controlling the application or service based upon the management information that is exposed when the application or service is installed on the system.

23. (Original) The method of claim 22, further comprising generating a manifest of the exposed management information.

24. (Original) The method of claim 22, the management information is exposed from one or more internal processes of a provider.

25. (Original) The method of claim 22, further comprising generating a catalog of all manifests of all available instrumentation data of the system, wherein the catalog is browsed and used to discover a particular instrumentation point.

26. (Original) The method of claim 22, the service is a native service whose code is wrapped with a managed code to facilitate attribution thereof.

27. (Original) The method of claim 22, the attributed code includes at least one of folder and probe attributes.

28. (Original) The method of claim 22, further comprising authoring the application or service with management information in preparation for a runtime.

29. (Original) The method of claim 22, further comprising generating an instrumentation manifest for the application or service based upon the management information.

30. (Original) The method of claim 29, the instrumentation manifest is stored with a collection of instrumentation manifests that are accessible to a consumer of the management information.

31. (Previously Presented) A system for managing an application or service, comprising:

- means for attributing selected parts of code of the application or service with health information;
- means for exposing the health information in the form of instrumentation definitions;
- means for cataloging the instrumentation definitions in a collection of instrumentation definitions; and
- means for controlling the application or service based upon the exposed instrumentation definitions when the application or service is installed on the system.

32. (Original) The system of claim 31, further comprising means for identifying the health information with a unique identifier.

33. (Previously Presented) A computer-readable medium having computer-executable instructions for performing a method for managing an application or service, the method comprising:

- receiving the application or service for installation on a system;
- attributing selected parts of code of the application or service with management information;
- exposing the management information to a management system; and
- controlling the application or service based upon the management information that is exposed when the application or service is installed on the system.

34. (Previously Presented) A computer-readable medium having computer-executable instructions that facilitates a system for managing an application or service, the system comprising:

an application or service for installation on the system; and

an attribution component that facilitates attributing selected parts of code of the application or service with management information;

wherein the system uses the management information to manage the installed application or service.

35. (Original) The computer-readable medium of claim 34, the management information includes a probe attribute that is used to indicate that a member of a type is a probe, which type is decorated with a folder attribute.

36. (Original) The computer-readable medium of claim 34, at runtime, a component of an in-process provider associated with the management information is loaded and invoked, and the management information for a decoupled provider is used with information provided by a register call to locate a corresponding running process, to connect to the running process, and to locate a subcomponent within the running process that is associated with the management information.

37. (Previously Presented) The system of claim 1, wherein the attributed parts of code are considered probes for use in determining health of the application.